Illiteracy

Illiteracy is a simple puzzle game. After the contest, if you'd like to play it, you can find it here: **https://le-slo.itch.io/illiteracy**. Of course, during the contest, it won't be accessible (and you've got better things to do!)

The game has a string of 8 icons. The icons in the game are very artistic, but for simplicity, we'll just call them **A**..**F**. Clicking any icon has a unique effect on the other icons. Most of the icons *Rotate* other icons. That means that they change **A→B**, **B→C**, **C→D**, **D→E**, **E→F**, and **F→A**.

There are 8 icon positions in a row, numbered left to right, 1 to 8. Here's what each of the icons do when clicked:

- **A**: *Rotates* the icon immediately to the left, if there is one, and immediately to the right, if there is one.
- **B**: If not on the end, changes the icon immediately to the right to be same as the one immediately to the left (does nothing on the ends). This is the only icon that doesn't *Rotate* other icons.
- **C**: *Rotates* the mirror image (when clicked in position **x**, *Rotates* 9-**x**. e.g. clicking 1 *Rotates* 8, 2 *Rotates* 7, etc.)
- **D**: *Rotates* all of the icons between this one and the closest end. (e.g. clicking 3 *Rotates* 1 and 2, 5 *Rotates* 6, 7 and 8. Clicking this icon on the end does nothing.)
- **E**: *Rotates* the closest end, and also the position which is the same distance in the opposite direction. (e.g. clicking 1 does nothing, 2 *Rotates* 1 and 3, 3 *Rotates* 1 and 5, 5 *Rotates* 8 and 2, 7 *Rotates* 8 and 6, etc.)
- **F**: *Rotates* another position with this pattern: Clicking 1 *Rotates* 5, 2 *Rotates* 1, 3 *Rotates* 6, 4 *Rotates* 2, 5 *Rotates* 7, 6 *Rotates* 3, 7 *Rotates* 8, and 8 *Rotates* 4. In other words, clicking an icon in an odd position *Rotates* (**x**+9)/2, even *Rotates* **x**/2.

Given a starting and target configuration, what's the smallest number of steps needed to get from the start to the target?

Input
Each input will consist of a single test case. Note that your program may be run multiple times on different inputs.

Each test case will consist of exactly two lines. Each line will have a string of length exactly 8, consisting only of the upper-case letters **A**, **B**, **C**, **D**, **E** and/or **F**. The first line holds the starting position, the second holds the target.

Output

Output a single integer, which is the smallest number of steps needed to get from the start to the target, or -1 if it isn't possible.

Sample Input	Sample Output
ABCDEFCD BBBBBBBB	9
BBBBBBBB ABCDEFCD	-1

Here is one possible way to solve the first case in the minimum 9 steps. The icon clicked is in **Inverse**, the icons which change are in **Bold**.

ABCDEFCD
ABDDEFCD
ABEDEFCD
BBEDFFCD
BBBDFFCD
BBBBFFCD
BBBBBFCD
BBBBBBCD
BBBBBBBD
BBBBBBBB